第5章 単3電池2本で2年駆動，LSI単価2ドルを目ざす無線通信規格“ZigBee”

メッシュ型トポロジでセンサ・ネットを実現

ワイヤレス・センサ・ネットワーク編

波多野祥二，水越剛成，水野将樹，山村和彦

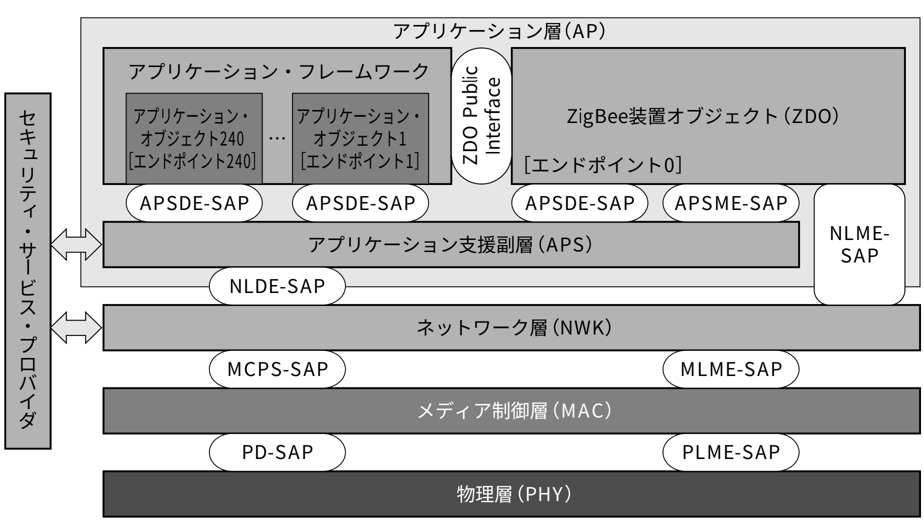
近距離無線通信規格「ZigBee」は，低消費電力で，かつ低コストの無線端末を実現するために策定された規格である．メッシュ型トポロジに対応していることからワイヤレス・センサ・ネットワークのインフラとして利用しやすい．ここでは，階層構造やトポロジ，プロトコル・スタック，プロファイルなど，基本的な項目について説明する．後半では，通信を行う際の要となるネットワーク層について詳細を述べる．（編集部）

ZigBeeの標準化団体であるZigBee Alliance(1)は，2004年12月にZigBee仕様 Ver.1.0を決定し，2005年6月末よりこの仕様を一般に公開しました．これにより，ZigBeeは本格的な普及に向けて前進を開始したと言えます．ここでは，まずZigBeeの概要を紹介し，続いてZigBeeの骨格を形成しているネットワーク層の動作の詳細を説明します．

● PANを低消費電力，低コストで実現することを目ざす

ZigBeeは，信頼性の高い低消費電力・低コストの近距離無線通信規格として4年ほど前からZigBee Allianceにて研究が進められてきました（右掲のコラム「ZigBeeの歴史とアライアンスの役割」を参照）．端末装置については，通信量を抑えればアルカリ単3電池2本で数ヵ月～2年間にわたり稼働できることを目ざしています．コストの面では，LSI単価で2ドル程度を目標としています．

ZigBee仕様で規定しているのはOSI参照モデルのネットワーク層以上の部分であり，物理層やMAC層についてはIEEE 802.15.4を採用しています（p.80のコラム「物理層とMAC層」を参照）．ZigBeeはPAN（personal area network）に分類されます．BluetoothやUWB（Ultra Wideband）の親戚筋にあたりますが，これらと比較してさらに



PD-SAP：物理層が上位層へ提供する物理層データの通信サービス用インターフェース
PLME-SAP：物理層が上位層へ提供する物理層管理サービス用インターフェース
MCPS-SAP：MAC層が上位層へ提供するMAC層データの通信サービス用インターフェース
MLME-SAP：MAC層が上位層へ提供するMAC層管理サービス用インターフェース
NLDE-SAP：ネットワーク層が上位層へ提供するネットワーク層のデータ通信サービス用インターフェース
NLME-SAP：ネットワーク層が上位層へ提供するネットワーク層管理サービス用インターフェース
APSDE-SAP：APS層が上位層へ提供するAPS層のデータ通信サービス用インターフェース
APSME-SAP：APS層が上位層へ提供するAPS層管理サービス用インターフェース
ZDO Public Interfaces：ZDOがアプリケーション・オブジェクトへ提供する，装置やネットワークを制御するためのインターフェース

図1　プロトコル・スタック(1)

ZigBeeのプロトコル・スタックは物理層（PHY），メディア制御層（MAC），ネットワーク層（NWK），アプリケーション層（AP）の四つの階層と，セキュリティ機能を担うセキュリティ・サービス・プロバイダ（SSP：Security Service Provider）からなる．

低消費電力，低コストの無線通信を実現することを目的として開発されました．

さらに，ネットワーク・トポロジとしてスター型，ツリー型（木構造），メッシュ型をサポートすることで，市場のさまざまな要求にこたえることもねらっています．

現在，照明制御，空調制御，物流管理，住宅制御，計測機器，センサ・ネットワークのインフラに，ZigBeeを搭載することを検討・評価している企業が徐々に増えています．ZigBeeは，これらの企業からの要求を反映させることで，標準規格としての成長を目ざしています．

1 ZigBeeの基本構造

図1に示すように，ZigBeeのプロトコル・スタックは物理層（PHY），メディア制御層（MAC），ネットワーク層（NWK），アプリケーション層（AP）の四つの階層と，セキュリティ機能を担うセキュリティ・サービス・プロバイダ（SSP：Security Service Provider）から構成されます．

● アプリケーション層は三つの層に分かれている

アプリケーション層は，アプリケーション支援副層（APS：Application Support Sublayer）とZigBee装置オブジェクト（ZDO：ZigBee Device Object），アプリケーション・フレームワークから構成されます．

アプリケーション支援副層のおもな機能は，認証とデータ転送とバインディングです．バインディングとは，サービスや要望に応じて二つの装置をつなげる機能です．

ZigBee装置オブジェクトはおもに，各層の初期化と装置/サービスの検索，セキュリティの可否決定，ZigBee装置種類の決定，APSのバインディング機能制御，ノード管理といった機能を提供します．

アプリケーション・フレームワークは，アプリケーション本体（一つのアプリケーション本体を「アプリケーション・オブジェクト」と呼ぶ）の動作環境です．アプリケーション・フレームワークは，アプリケーション・オブジェクトを最大240個持つことができます．また，アプリケーション・フレームワークはアプリケーション・オブジェクトに対して二つのデータ通信サービスを提供します．一つは

Column ZigBeeの歴史とアライアンスの役割

ここでは，ZigBeeの歴史について触れてみたいと思います．

2000年ごろから，ZigBeeのようなネットワークの実現を望む声が技術者から聞かれるようになりました．理由は，ワイヤレスLAN（WiFi）やBluetoothではコストや消費電力の面で問題があり，実現できないシステムがあったためだと思います．

その後，**表A-1**のような経緯をたどり，現在に至っています．

ZigBee Allianceは，オープンな世界標準に基づいた，信頼性が高く，低コストで，低消費電力のワイヤレス・ネットワーク製品を提供するために設立された非営利団体です．ZigBee Allianceのメンバ構成と会員数は，Promoter会員が8社，Participant会員が133社，Adopter会員が31社です（2005年8月3日現在）．

Promoter会員（年会費40,000ドル）は，Allianceのボード・メンバとしてすべての権利が認められています．Participant会員（年会費9,500ドル）はPromoter会員とほとんど同等の権利が与えられています．ただし，ボード・メンバになれない，仕様を批准できない，ワーク・グループの議決権を持てないなどの違いがあります．Adopter会員（年会費3,500ドル）は，完成したZigBeeの仕様を受け取ること，イベントやワークショップに参加できることなどが認められています．

ZigBee Alliance設立時のPromoter会員は，米国Honeywell International社，三菱電機，米国Motorola社，オランダRoyal Philips Electronics社の4社でしたが，現在はノルウェーのChipcon社，米国Ember社，米国Freescale Semiconductor社，韓国Samsung Electronics社が加わり，合計8社となりました．日本企業としては，**表A-2**に示す合計16社がZigBee Allianceに参加しています．

ちなみに，“ZigBee”という名まえの由来ですが，実は，Alliance内でも異説が多々存在しています．「はち（bee）がジグザグ（Zig Zag）に動きながら，花やはちみつの場所を仲間に知らせる方法」を「ZigBee Principle（ZigBee原理）」と呼ぶことからこの名がつけられたとか，「くもの巣説（くもの巣のように空間に無線通信網を張る）」などが由来として広く知られているようです．

表A-1 ZigBee仕様ができ上がるまで

年 月	活動内容
2002年10月	ZigBee Alliance設立発表
2002年12月	ZigBee Alliance会員数36社
2003年 5月	IEEE 802.15.4作成完了
2004年12月	ZigBee Alliance会員数113社， ZigBee Specification Ver.1.0作成完了
2005年 6月末	ZigBee Specificationの一般公開を開始

表A-2 ZigBee Allianceに参加している日本企業

Promoter会員	三菱電機
Participant会員	アキタ電子システムズ，NECエンジニアリング，沖電気工業，OTSL，オムロン，新光電気工業，セイコーエプソン，村田製作所，安川情報システム，山武，横河電機，ルネサス テクノロジ
Adopter会員	三技協，シイガイズ，ステップワン

属性値を扱えるサービス「Key Value Pair(KVP)Service」であり，もう一つは汎用的なメッセージ・サービス「Message Service」です．

● **ネットワークは三つの「装置」で構成される**

次に，「ZigBee装置」がどういったものなのかを述べておきます．ZigBee装置は，以下の三つに分類されます．

- **コーディネータ(coordinator)**――コーディネータは，ZigBeeネットワーク全体を管理する装置．一つのネットワークに一つだけ存在する
- **ルータ(rooter)**――ルータは，データを中継する機能を提供する装置
- **エンド・デバイス(end device)**――エンド・デバイスは，ネットワーク構造上の末端に位置づけられることが多い．データを中継する機能は持たないが，使用する資源を少なく抑えることができるので，低コストと低消費電力を実現できる

これら3種類のZigBee装置を用途に応じて使い分けて，無線によるネットワークを構成します．

ZigBee(IEEE 802.15.4)は，約65,000個のノードからなるネットワークを構成できます．ネットワーク・トポロジは，スター型，ツリー型，メッシュ型に対応しています(**図2**)．メッシュ型ネットワークでは，ルータどうし，またはコーディネータの間を直接通信できます．これによって，比較的大きなネットワークを容易に構築できます．

2 ZigBeeのプロファイル

ZigBee AllianceはZigBeeの用途として，以下のものを想定しています．

- 住宅制御(Home Control)
- ビルディング自動化(Building Automation)
- 工業自動化(Industrial Automation)

互換性のあるシステムを実現するためには，用途ごとにプロファイルを規定する必要があります．ZigBeeにおけるプロファイルとは，アプリケーションとしての通信データのフォーマットと各ZigBee装置のアプリケーション機能を記述したものを指します．プロファイルは，ZigBeeとして認定されるもっとも基本的な要件の一つです．

● **プロファイルは3種類**

プロファイルとしては，次の三つが存在します．

1) Public

Publicプロファイルは，ZigBee Allianceが策定するプロファイルです．すでに，Home Controls(0x01)，Building Automation(0x02)，Plant Control(0x03)の三つのプロファイルが規定されています(カッコ内はプロファイル識別子を表す)．

2) Publish

Publishプロファイルは，Publicではないが，プロファ

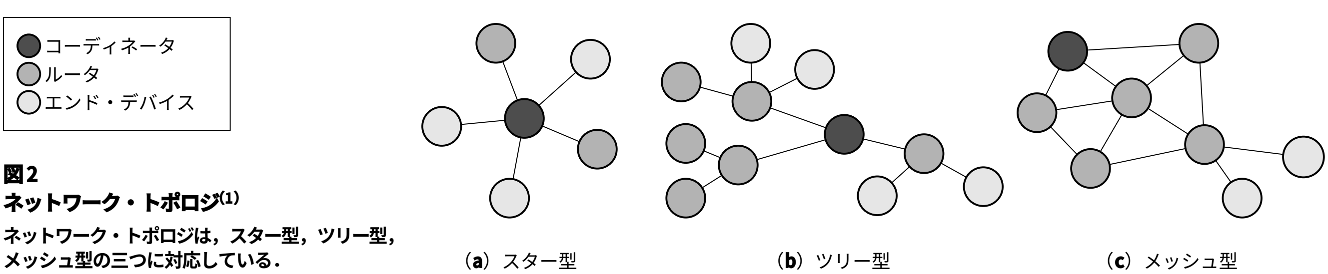


図2
ネットワーク・トポロジ(1)
ネットワーク・トポロジは，スター型，ツリー型，メッシュ型の三つに対応している．

Column 物理層とMAC層

ZigBeeは，物理層およびMAC層にIEEE 802.15.4を使用します．

利用可能な周波数帯として，868MHz帯，915MHz帯，2.4GHz帯の3種類が定義されており，日本では2.4GHz帯を利用できます．

2.4GHz帯の最大データ転送速度は250kbpsであり，通信距離は10～100m程度です．スペクトラム拡散方式はDS-SS(直接拡散方式)であり，アクセス制御方式にCSMA/CA(carrier sense multiple access with collision avoidance)を使用するところは，ワイヤレスLAN規格のIEEE 802.11bに似ています．また変調方式は2.4GHz帯ではQPSK(quadrature phase shift keying)を，915/868MHz帯ではBPSK(bi-phase shift keying)を利用します．

現在，IEEE 802.15.4の改訂版であるIEEE 802.15.4bの仕様策定が進んでおり，将来ZigBeeにも組み込まれていくでしょう．

イルの内容が公開されるものです．

3）Private

Privateプロファイルは，ZigBee Allianceのメンバが独自に作成し，利用するものです（公開の義務はない）．

いずれのプロファイルも，それを作成するにはZigBee Allianceからプロファイル識別子（2バイトのデータ）を取得する必要があります．

● 端末開発にはIEEE 802.15.4準拠の無線ハードが必須

プロファイルを含め，ZigBee仕様に準拠した端末装置を開発するには，IEEE 802.15.4に準拠した近距離無線ハードウェアが必要になります．すでにノルウェーのChipcon社や米国Freescale Semiconductor社からは，ZigBee用途向けのRF LSIおよびプロトコル・スタック・ソフトウェアが発売されています．また，アキタ電子システムズ，NECエンジニアリング，新光電気工業などからは，RF LSIとマイクロプロセッサ，アンテナなどの周辺回路を一つの基板に載せたモジュールも発表されています．

3 ZigBeeのネットワーク層

ZigBeeのネットワーク層の特徴として，スター型，ツリー型，メッシュ型といったさまざまなトポロジに対応できることが挙げられます．とくに，メッシュ型に代表される無線によるマルチホップ通信により，直接通信できない相手ともルータを介して無線ネットワークを構築できます（p.82のコラム「孤立からの復旧」を参照）．

ネットワーク層では，コーディネータがネットワークを開始し，そこにルータやエンド・デバイスが参加（join）してネットワークを構築します注．また，すでにネットワークに参加しているルータを経由して別のルータやエンド・デバイスがネットワークへ参加できます．コーディネータと直接通信できなくてもネットワークに参加できるので，結果として規模の大きなネットワークができ上がります．

以下に，ネットワークに参加する際に新しいアドレス（ショート・アドレス）を割り振ってネットワークを構築する方法，および構築されたネットワークにおけるマルチホップによるデータ通信を行う方法を解説します．

● アドレスを割り当てる方法は2種類

ZigBeeでは，コーディネータのショート・アドレスはつねに‘0’を使用します．ルータとエンド・デバイスは，ネットワークに参加する際に，ショート・アドレスが割り当てられます．

注：ネットワークの開始とは，最初のネットワーク，すなわちネットワークを構成するZigBee装置が一つしかない状態において，その装置がコーディネータになることを宣言すること．ネットワークの構築とは，コーディネータに対してほかのZigBee装置がネットワークへの参加を要求し，コーディネータがこれにこたえること．

Column 電波干渉の話

ある部屋においてワイヤレスLANが頻繁に使われている状態でZigBeeのネットワークを動作させる場合，ZigBee装置は次のようにふるまいます．

1）利用可能な16チャネルの中から指定した複数チャネルの電波強度を測定する．
2）1）の結果，空いていると判断できるチャネルを選んで，ネットワークを動作させる（コーディネータの開始）．
3）コーディネータが動的にチャネルを決定した場合，ネットワークに参加を希望する装置は，どのチャネルに参加したいネットワークが存在するかを調べる．
4）3）の結果，装置は目的のチャネル情報を得てそのチャネルを使用する．

このような手順を踏むことで，ZigBeeはワイヤレスLANなどとの衝突を避けながら，起動します．

ただし，起動時に選んだチャネルが将来にわたって空いたままであるという保証はありません．無線機器どうしの競合の原因の一つが，このことに起因しています．

IEEE 802.15.4（ZigBeeの物理層）では，通信の途中でコーディネータがチャネルを変更し，その子である装置たちが新しいチャネルで再構築される機能（Orphan機能；迷子の呼び出しみたいなもの）が提供されています．ZigBeeのネットワーク層などでこれらのしくみをうまく使えば，競合の問題もうまく解消できるでしょう．

しかし，ZigBee 1.0の仕様には，いったん動き出した後，衝突をどうやって自動的に回避するかという点は明記されていません．このため，この処理はアプリケーションでくふうすることになります．

具体的には，アプリケーションからネットワークを再起動する，あるいは，アプリケーションからOrphan機能を利用してチャネルを変更する，といったことが考えられます．このようにして，その時点で空いているチャネルを選ぶことで，衝突を回避することになります．ただし，Bluetoothでもそうであったように，衝突回避については将来なんらかの仕様がZigBee Allianceから提示される可能性は高いと筆者は考えています．

装置にアドレスを割り当てる方法は2種類あります．一つはネットワーク層で自動的に割り当てる方法，もう一つは上位層から割り当てるアドレスを指定する方法です．それぞれの方法を，「分散アドレス割り当てメカニズム」，「上位層アドレス割り当てメカニズム」と呼びます．

1）分散アドレス割り当てメカニズム

この方法は，あらかじめ指定されたネットワークの構造情報をもとにZigBeeのネットワーク層で自動的にアドレスを割り当てます．この方法がデフォルトの動作になります．

ZigBeeのコーディネータに，ネットワークの構造情報として，「分散アドレス割り当てパラメータ」の設定を行います（**表1**）．ZigBeeのネットワーク層は，この情報をもとに，木構造でアドレスを割り当てます．**図3**に分散アドレス割り当ての例を示します．

各装置に割り当てるアドレスの計算は，次のように行います（**図4**）．コーディネータには，アドレス0を割り当てます．指定されたパラメータ（Cm，Rm，Lm）から，木構造の深さ（d）に応じたアドレス・ブロックの大きさ$Cskip(d)$を計算します．

表1　分散アドレス割り当てパラメータ

パラメータ	記　号	内　容
nwkMaxChildren	Cm	子デバイスの最大接続数（ルータとエンド・デバイスの合計）
nwkMaxDepth	Lm	ネットワークの最大の深さ（コーディネータから末端のデバイスまでのホップ数）
nwkMaxRouters	Rm	ルータの最大接続数

$$Cskip(d)=\begin{cases}1+Cm\cdot(Lm-d-1) & Rm=1\\ \dfrac{1+Cm-Rm-Cm\cdot Rm^{Lm-d-1}}{1-Rm} & Rm\neq 1\end{cases}\quad\cdots(1)$$

接続要求を行ったルータに対しては，要求した順（$n=1$，…，Rm）にアドレスA_nを割り当てます．

$$A_n=A_{parent}+1+Cskip(d)\cdot(n-1)\quad\cdots\cdots\cdots\cdots(2)$$

ここで，A_{parent}は親のアドレス

接続要求を行ったエンド・デバイスについて，要求した順（n=1，…，$Cm-Rm$）にアドレスA_nを割り当てます．

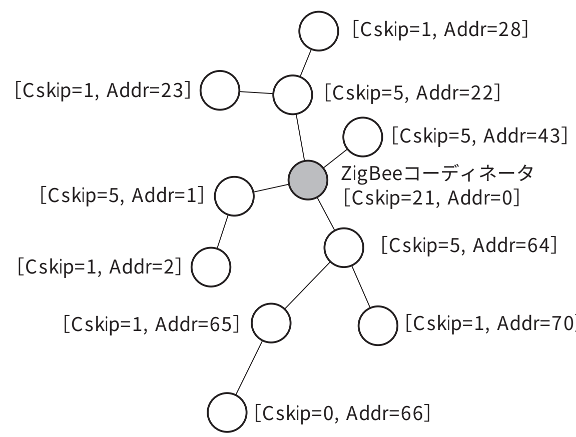


図3　分散アドレス割り当ての例（Cm＝4，Rm＝4，Lm＝3の場合）(1)

Cm＝4，Rm＝4，Lm＝3の場合のアドレス割り当ての例を示す．通信経路は線で示す．コーディネータは，参加要求を行ったルータに対して，アドレス1，22，43，64を割り振っている．

Column　孤立からの復旧

「ZigBeeは，直接通信できない相手とルータなどを介して通信できる」と本文で述べました．これはZigBeeの重要な特徴の一つですが，逆に言うと，ルータや（とくに）コーディネータでトラブルが発生したり，利用していた周波数帯がほかの無線の影響で使えなくなった場合，だれとも通信できず孤立する可能性があるとも言えます．この場合，孤立した状態から再度ネットワークに加わる必要がありますが，この復旧プロセスはアプリケーションに任されています．

トラブルの典型的な例として，停電などによる電源遮断が挙げられます．この場合，コーディネータ/ルータの管理情報や参加している子装置の情報が不揮発メモリに残っていれば，IEEE 802.15.4のOrphan機能によって復旧できます．“Orphan”は直訳すると「孤児」となるわけですが，電源復旧後，孤立したルータやエンド・デバイスがOrphan Scanを行うと，以前親であった装置が元のショート・アドレスやPAN IDなどのネットワーク参加に必要な情報を装置に返して，ネットワークを復旧することができます．不揮発メモリに情報を残していない場合は，もう一度ネットワーク参加（join）を行うことで，ネットワークを再構築します．ただし，このケースでは以前のネットワークと異なるショート・アドレスが割り当てられる可能性があるため，トラブル発生前とまったく同じ状態には戻りません．

トラブルのもう一つの例として，ワイヤレスLANなどによる妨害があります．この場合，コーディネータが利用するチャネルを変えることで対処します．チャネルを変える際，新しいチャネルはブロードキャストで通達されますが，これを受信できなかったルータや通常受信状態にないエンド・デバイスなどは，チャネルの変更情報を得ることができません．このようなケースでは，上述のOrphan Scanを複数のチャネルに対して実行することで，新しいチャネルにおけるネットワークの復旧が可能となります．

なお，Orphan ScanはMAC層の機能です．これをネットワーク層経由で利用するには，ネットワークへの参加機能（join）のオプションを用いて再参加（rejoin）を指定します．

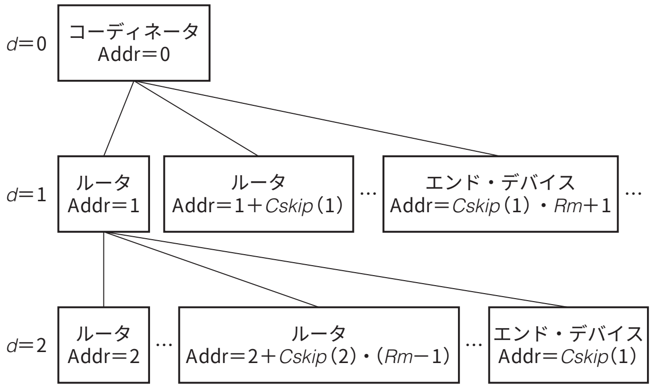


図4　分散アドレス割り当てメカニズム
コーディネータやルータは，参加要求を行ったルータに対して自アドレス+1から *Cskip* ごとにアドレスを割り当てる．エンド・デバイスに対しては，ルータ用アドレス空間の後ろのアドレスを続けて割り当てる．

$$A_n = A_{parent} + Cskip(d) \cdot Rm + n \quad \cdots\cdots(3)$$

2）上位層アドレス割り当てメカニズム

この方法は，上位のプロトコル層（例えばアプリケーションなど）からZigBeeのネットワーク層に対して割り当てるアドレス空間を指定することによって，アドレスの割り当てを行います．

ここで使用するパラメータは，「上位層アドレス割り当てパラメータ」です（**表2**）．

ZigBeeのネットワーク層の動作としては，nwkAvailableAddressが0より大きい場合，ネットワークへの参加を許可します．ほかの装置から新たにネットワークへの参加要求が行われた場合，nwkNextAddressで指定されているアドレスを割り当てます．その後，nwkNextAddressの値をnwkAddressIncrementの値だけ増やし，nwkAvailableAddressを一つ減らします．

● 通信品質の良い経路を選択するアルゴリズムを用いる

ZigBeeでは，直接通信できない装置どうしであっても，ルータがデータ・フレームを中継することにより通信が可能です．また，メッシュ型トポロジではルータ間の直接送信が許されているため，ある装置から別の装置へデータを送信する経路が複数存在します．このため，効率の良い経路を選択するルーティングが非常に重要となります．

ここでは，ZigBeeのルーティングの概略を，パス・コスト距離（ルーティング・コスト），データ・フレーム中継（ルーティング・テーブル），ルート検索の三つの観点から説明します．

表2　上位層アドレス割り当てパラメータ

パラメータ	内　容
nwkNextAddress	次に割り当てを行うアドレス
nwkAvailableAddress	割り当てを行うデバイス数
nwkAddressIncrement	割り当てを行うアドレス値の間隔

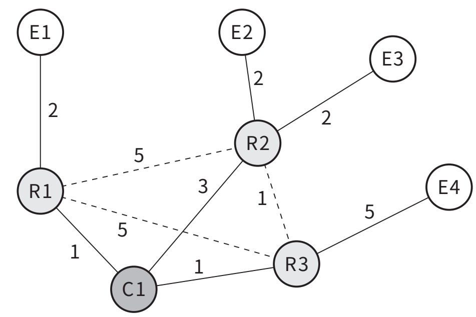


図5　パス・コスト距離
この図のネットワークは，コーディネータが1個（C1），ルータが3個（R1～R3），エンド・デバイスが4個（E1～E4）で構成されている．それぞれを結ぶ線上に書かれている数字が各リンク・コストを表す．

1）ルーティング・コスト

ネットワーク層では，通信経路を決定するルーティング・アルゴリズムとして，パス・コスト距離（path cost metric）を利用します．パス・コスト距離は，ある経路における各ルータ間のリンク・コストの合計です．これを計算式で書くと，次のようになります．

$$C\{P\} = \sum_{i=1}^{L-1} C\{[D_i, D_{i+1}]\} \quad \cdots\cdots(4)$$

ここで，P＝パス，L＝パスの数，
$C\{[D_i, D_{i+1}]\}$＝装置Di と D_{i+1} の間のリンク・コスト

リンク・コストとは，二つの装置（例えばルータとルータ）の間の通信品質のようなもので，一般にはIEEE 802.15.4のMAC層から得られるデータ受信時の LQI（link quality indication）を利用して算出します．リンク・コストは0～7の数値で表現され，数が大きいと通信にかかるコストが大きい，つまり通信品質が悪いことを意味します．したがって，通常，装置間の距離が長ければ大きくなります．また，妨害電波が多くても，大きくなります．

つまり，パス・コスト距離とは，各装置間の通信品質（リンク・コスト）を距離に見立て，もっとも距離の短い経路を選択するアルゴリズムです．

図5のようなメッシュ型のPANがあるとします．この例では，コーディネータが1個（C1），ルータが3個（R1～R3），エンド・デバイスが4個（E1～E4）でネットワークが構成されています．それぞれを結ぶ線上に書かれている

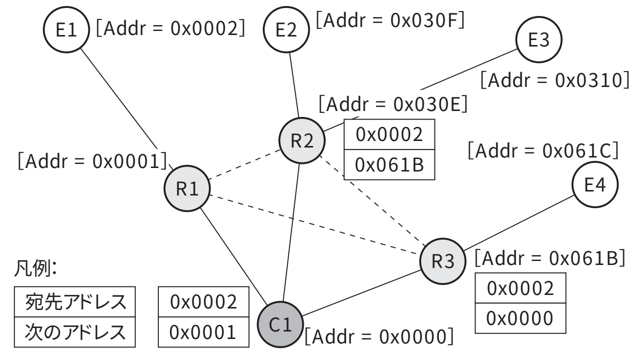


図6　ルーティング・テーブル
各ルータ(R2，R3，C1)はアドレス0x0002に対するルーティング・テーブルを持っている．このテーブルに従ってE3→R2→R3→C1→R1→E1という経路でデータ・フレームが転送される．

数字が各リンク・コストを表します．

このPANにおいて，E3からE1へデータを送信したい場合，五つの経路が考えられます．もっとも転送回数が少ないのはE3→R2→R1→E1という経路ですが，この場合のリンク・コストの合計は9となります．一方，E3→R2→R3→C1→R1→E1という経路をとった場合，転送回数は多くなるものの，リンク・コストの合計は7となります．すなわちこのPANでは，E3からE1へデータを送信するには，リンク・コストが小さいこちらの経路が選択されます．

2）ルーティング・テーブル

選択した通信経路はルータ内にルーティング・テーブルとして保存され，データ送信の際に利用します(なお，ここではとくに分けて書いていないかぎり，「ルータ」はルータおよびコーディネータを意味する)．

ルーティング・テーブルは，宛先アドレス(destination address)へ到達するために次に送信すべきアドレス(next-hop address)のペアを要素とするテーブルです．ルータがデータ・フレームを中継する際，宛先アドレスをもとにルーティング・テーブルを検索して見つけた次の送信先へ送信します．各ルータで，この処理を繰り返すことによって最終的に宛先アドレスにデータ・フレームが到達します．

例えば，**図6**のPANにおいて，エンド・デバイスE3(アドレス＝0x0310)からエンド・デバイスE1(アドレス＝0x0002)に対してデータを送ろうとした場合，各ルータはアドレス0x0002に対するルーティング・テーブルを持っているため，このテーブルに従ってE3→R2→R3→C1→R1→E1という経路でデータ・フレームが転送されます．

3）ルート検索

はじめからルーティング・テーブルが存在しているのであれば話は簡単ですが，実際には状況に応じて構築する必要があります．

ネットワーク層は，上位層からの要求，もしくは受信したフレーム内の情報によってルート検索を指示された場合，ルート検索を開始し，ルーティング・テーブルを構築します．

ZigBeeのルート検索は，ブロードキャストで宛先アドレスを含んだroute requestコマンドを送信することによって実現します．このコマンドは近傍だけではなく，指定されたホップ数の分だけ転送されます．ルータはこのコマンドを転送する際に，コマンドに含まれる送信元からのコストの合計と直前の送信者をルート検索テーブルに記録しておき，応答を返す際の応答先の決定に利用します．

route requestコマンドが宛先アドレスまたはその親までたどり着くと，逆の順路をたどって応答(route responseコマンド)が返信されます．各ルータは応答を受信(転送)する際に，自身のルーティング・テーブルを構築し，結果としてルートが確立します．このとき，ルータは複数の経路から応答を受け取る可能性があります．応答には宛先アドレスまでのリンク・コストの合計が含まれるので，この値をもとにパス・コスト距離に従って，ルーティング・テーブルの登録内容を決定します．

参考・引用*文献

(1)* ZigBee Allianceのホームページ，http://www.zigbee.org/

はたの・しょうじ，みずこし・たかなり，みずの・まさき，やまむら・かずひこ　　(株)OTSL

<筆者プロフィール>

波多野祥二．2003年4月にOTSLを8名で設立．代表取締役兼CEOに就任．米国Figure 8 Wireless社と提携してZigBeeを日本国内へ普及させる活動を行っている．

水越剛成．OTSL設立メンバ．UNIXのデバイス・ドライバやカーネルの開発をはじめ，組み込み向けコンパイラ，デバッガなどの開発を経て，最近はBluetooth，ZigBeeなどの無線通信系の開発に従事．なぜか，年数を経るごとに使用するCPUのビット数が減ってきている．

水野将樹．OTSL設立メンバ．UNIXのデバイス・ドライバの開発から，カーネル，TCP/IPプロトコル・スタック(IPv4)，組み込み向けの開発などを行う．社内ネットワーク管理を行いながら，オシロスコープを片手に逆アセンブルしたコードと格闘するソフトウェア技術者．

山村和彦．OTSL設立メンバ．1986年ごろから8086アセンブラにはまり，組み込みの世界へ．途中，VICSやETCのシステム開発を経て現在に至る．ふだんは営業・経理を担当．CEOの悪だくみによって，急にエンジニアとして売りに出されることも….